TITLE OF THE INVENTION

METHOD AND APPARATUS FOR CONTROLLING AMOUNT OF DCT COMPUTATION PERFORMED TO ENCODE MOTION IMAGE

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the benefit of Korean Application No. 2002-40405, filed July 11, 2002, in the Korean Intellectual Property Office, the disclosure of which is incorporated herein by reference.

BACKGROUND OF THE INVENTION

1. Field of the Invention

[0002] The present invention relates to a motion image encoding method and apparatus, and more particularly, to a method and apparatus to control the amount of DCT computation.

2. Description of the Related Art

[0003] FIG. 1 is a block diagram showing an encoder 120 and a decoder 140 which are used for general motion image encoding. To achieve video on demand (VOD) services or motion image communications, the encoder 120 produces an encoded bit stream using a compression technique, and the decoder 140 restores an image from a received bit stream.

[0004] A discrete cosine transform (DCT) unit 122 performs a DCT operation on received image data in units of 8×8 pixel blocks to remove spatial correlation. A quantization (Q) unit 124 performs quantization on a DCT coefficient obtained by the DCT unit 122 to accomplish a high efficient loss-compression.

[0005] An inverse quantization (IQ) unit 126 inversely quantizes image data quantized by the Q unit 124. An inverse DCT (IDCT) unit 128 performs an IDCT on image data inversely quantized by the IQ unit 126. A frame memory 130 stores image data IDCT-ed by the IDCT unit 128, on a frame-by-frame basis.

[0006] A motion estimation (ME)/motion compensation (MC) unit 132 estimates a motion vector (MV) for individual macro blocks and a sum of absolute difference (SAD) corresponding to a block matching error, based on a received current image data frame and a previous image data frame that is stored in the frame memory 130.

[0007] A variable length coding (VLC) unit 134 removes statistical redundancy from DCT-ed and quantized data using the MV estimated by the ME/MC unit 132.

[0008] A bit stream encoded by the encoder 120 is decoded by the decoder 140, which includes a variable length decoding (VLD) unit 142, an IQ unit 144, an IDCT unit 146, a frame memory 148, and an MC unit 150.

[0009] VOD services or real-time motion image communications require real-time motion image encoding and decoding, which are achieved by effectively controlling the amount of computation made by an encoder and a decoder. In most motion image encoding techniques, including MPEG 4 and H.26x, a DCT unit and an ME unit require the greatest amount of computation.

[0010] Hence, the encoder uses a DCT skipping technique and a fast ME algorithm to reduce the amount of computation made by the DCT unit and the ME unit. The amount of computation in the decoder is determined depending on the result of encoding by the encoder and is affected by the number of skipped blocks.

[0011] In addition, United States Patent Publication No. 6,122,321 discloses an encoder similar to the apparatus shown in FIG. 1.

[0012] FIG. 2 shows a motion image encoder to which a conventional DCT skipping technique is applied. The structure of the motion image encoder of FIG. 2 is obtained by adding a DCT computation amount controller 280 to control the amount of DCT computation made by a DCT unit 210, to the encoder 120 of FIG. 1.

[0013] The DCT computation amount controller 280 compares an SAD for each block estimated by an ME 270 and a quantization coefficient (Q) of a Q unit 220 with a predetermined threshold value (T) to determine whether a DCT performed on each 8x8 pixel block in a DCT unit 210 is to be skipped. To be more specific, if the SAD/Q is smaller than the predetermined threshold value T, an end of block (EOB) is highly likely to be nearly 0, and accordingly the DCT computation amount controller 280 does not process received data. On the other hand, if the SAD/Q is greater than the predetermined threshold value T, the DCT computation amount controller 280 controls a DCT to be performed. Consequently, the amount of a DCT computation performed in the encoder is reduced.

[0014] However, in the case of conventional motion image encoders, there is a limit in the amount of computation that can be performed in a decoder, and the fact that an allocated computation amount and an actual computation amount may vary depending on circumstances is not considered. Hence, when a bit stream encoded by a conventional motion image encoder is input, the decoder cannot properly encode the received bit stream or cannot make full use of the allocated computation amount.

[0015] Also, in the case of conventional motion image encoders, the predetermined threshold value T is a fixed value within an image sequence, which means that the fact that DCT computation complexity varies for each macro block or frame is not considered. Thus, real-time encoding of received motion image data is not properly performed.

SUMMARY OF THE INVENTION

[0016] The present invention provides a method and apparatus to control the amount of DCT computation performed to encode motion images in an encoder in consideration of the amount of computation in a decoder, so that a computation amount allocated to the decoder is fully used, and an input bit stream is properly encoded.

[0017] The present invention also provides a method and apparatus to control the amount of DCT computation performed to encode motion images, in which the computation complexity of a motion image encoder is kept constant regardless of the characteristics of images by applying a variation in the amount of motion estimation computation by a motion estimation unit to a DCT unit.

[0018] According to an aspect of the present invention, a method controls the amount of discrete cosine transform (DCT) computation to encode motion images. In the method, decoder complexity information that represents the amount of decoding computation performed on previous image data is received. Then, the skipping of DCT operations on current image data is controlled based on the received decoder complexity information.

[0019] Generally, the decoder complexity information receiving step further includes calculating the decoder complexity information which represents the amount of decoding computation performed on previous image data.

[0020] Typically, the DCT skipping controlling includes: calculating a target DCT computation amount of the current image data using the received decoder complexity information; calculating a threshold value used to skip execution of a DCT on the current image data, based on the calculated target DCT computation amount; and controlling the skipping of DCT operations on the current image data, depending on the calculated threshold value.

[0021] According to an aspect of the present invention, a method controls the amount of DCT computation to encode motion images. In the method, both decoder complexity information that represents the amount of a decoding computation on previous image data and ME computation amount variation information that represents information on a variation in the amount of a motion estimation (ME) computation are received. Then, the skipping of DCT operations on current image data is controlled based on the received decoder complexity information and the received ME computation amount variation information.

[0022] Generally, the DCT skipping controlling includes: calculating a target DCT computation amount of the current image data using the received decoder complexity information and the received ME computation amount variation information; calculating a threshold value used to skip execution of an DCT on the current image data, based on the calculated target DCT computation amount; and controlling the skipping of DCT operations on the current image data, depending on the calculated threshold value.

[0023] According to another aspect of the present invention, an apparatus controls the amount of a discrete cosine transform (DCT) computation to encode motion image. The apparatus includes a DCT computation amount controller and a DCT unit. The DCT computation amount controller controls the skipping of DCT operations on current image data using decoder complexity information which represents the amount of decoding computation on input previous image data. The DCT unit performs a DCT operation or skips execution of a DCT operation on the input present image data based on a control signal output from the DCT computation amount controller.

[0024] Preferably, the apparatus further includes a decoder complexity calculator which calculates the decoder complexity information that represents the amount of a decoding computation on input previous image data.

[0025] Also, typically, the DCT computation amount controller calculates a target DCT computation amount of the current image data using the decoder complexity information, calculates a threshold value used to skip execution of a DCT operation on the current image data, based on the calculated target DCT computation amount, and controls the skipping of DCT operations on the current image data depending on the calculated threshold value.

[0026] According to another aspect of the present invention, an apparatus controls the amount of a DCT computation to encode motion images. The apparatus includes a DCT computation amount controller and a DCT transformer. The DCT computation amount controller controls the skipping of DCT operations on current image data using both decoder complexity information which represents the amount of decoding computation on previous image data and ME computation amount variation information which represents information on a variation in the amount of motion estimation (ME) computation. The DCT transformer performs a DCT or skips execution of a DCT on the input present image data based on a control signal output from the DCT computation amount controller.

[0027] Generally, the DCT computation amount controller calculates a target DCT computation amount of the current image data using the decoder complexity information and the ME computation amount variation information, calculates a threshold value used to skip execution of a DCT operation on the current image data, based on the calculated target DCT computation amount, and controls the skipping of DCT operations on the current image data depending on the calculated threshold value.

[0028] Additional aspects and advantages of the invention will be set forth in part in the description which follows and, in part, will be obvious from the description, or may be learned by practice of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

[0029] These and/or other aspects and advantages of the invention will become apparent and more readily appreciated from the following description of the preferred embodiments, taken in conjunction with the accompanying drawings of which:

FIG. 1 is a block diagram showing an encoder 120 and a decoder 140 which are used to perform a general motion image encoding as is known in the art;

FIG. 2 is a block diagram of a motion image encoder employing a conventional discrete cosine transform (DCT) skipping technique;

FIG. 3 is a block diagram showing a motion image encoder according to an embodiment of the present invention and a decoder;

FIG. 4 is a flowchart illustrating a method of controlling the amount of a DCT computation, according to an embodiment of the present invention;

FIG. 5 is a block diagram of a motion image encoder according to another embodiment of the present invention and a decoder; and

FIG. 6 is a flowchart illustrating a method of controlling the amount of a DCT computation, according to another embodiment of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0030] Reference will now be made in detail to the present preferred embodiments of the present invention, examples of which are illustrated in the accompanying drawings, wherein like reference numerals refer to the like elements throughout. The embodiments are described below in order to explain the present invention by referring to the figures.

[0031] Referring to FIG. 3, a motion image encoder 320 according to the present invention includes a DCT unit 322, a Q unit 324, an IQ unit 326, an IDCT unit 328, a frame memory 330, an ME unit 332, a VLC unit 334, and a DCT computation amount controller 336. The DCT unit 322 performs a DCT computation on received image data. The Q unit 324 performs a quantization on a DCT coefficient obtained by the DCT unit 322, using a quantization coefficient Q. The IQ unit 326 inversely quantizes image data quantized by the Q unit 324. The IDCT unit 328 performs IDCT on image data inversely-quantized by the IQ unit 326. The frame memory 330 stores image data IDCT-ed by the IDCT unit 328 on a frame-by-frame basis. The ME 332 produces motion estimation related information using a currently received image data frame and a previous image data frame stored in the frame memory 330. The VLC unit 334 performs encoding by removing statistical redundancy from the DCT-ed and quantized data. The DCT computation amount controller 336 controls the amount of the DCT computation made by the DCT unit 322 based on decoder complexity information output from a decoder complexity calculator 342.

[0032] The decoder 340 includes a VLD unit, an IQ unit, an IDCT unit, a frame memory, and an MC unit (not shown), which correspond to units of the same designation of the decoder 140 of FIG. 1. Thus, the structure of the decoder 340 will not be described in detail.

[0033] When the decoder 340 has completely decoded a previous frame, that is, an n-th frame (hereinafter, referred to as f(n)), the decoder complexity calculator 342 produces information on the amount of decoding computation performed on the previous frame f(n), that is, decoder complexity information, and transmits the decoder complexity information via a backward channel to the DCT computation amount controller 336 included in the encoder 320. If a ratio of the amount of processing of an IDCT computation to the amount of the decoding computation on the previous frame f(n) is $C_{IDCT}(n)$, the amount of computation that may be performed by the decoder is $C_{allowed}(n)$, and the actual amount of computation is $C_{total}(n)$, and decoder complexity information for k recently-decoded frames is $\{ C_{IDCT}(j), C_{allowed}(j)\ C_{total}(j) |\ j= n-k+1, \ldots, n\}$. Here, k denotes a variable that may be set selectively.

[0034] The actual amount of computation $C_{total}(n)$ denotes the total amount of computation performed on the n-th frame by the VLD unit, the IQ unit, the IDCT unit, and the MC unit included in the decoder 340. Although the IDCT computation amount $C_{IDCT}(n)$ is included in the decoder complexity information, an IDCT computation amount may be obtained by the IDCT unit 328 of the encoder 320, instead of including the IDCT computation amount $C_{IDCT}(n)$ in the decoder complexity information.

[0035] The DCT computation amount controller 336 controls the number of DCT operations performed on a current frame f(n+1) to be encoded among received image data in the DCT unit 322 according to a DCT skipping technique, based on the decoder complexity information received from the decoder complexity calculator 342.

[0036] In an embodiment of the present invention, if the actual amount of computation $C_{total}(n)$ of the previous frame f(n) decoded by the decoder 340 is smaller than the computation amount allowed to the decoder 340, $C_{allowed}(n)$, the decoder does not currently make full use of allowed resources. Accordingly, the number of skipped DCT operations to be performed on frames next to a current frame f(n+1) is reduced to increase the amount of computation made by the encoder 320 and the decoder 340. In this case, the decoder 340 may properly perform decoding, and good image quality may be provided because of a reduction in the number of skipped DCT operations.

[0037] On the other hand, if the actual amount of computation $C_{total}(n)$ of the previous frame f(n) decoded by the decoder 340 is greater than the amount of computation that may be performed by the decoder 340, $C_{allowed}(n)$, the number of skipped DCT operations to be performed on frames next to the current frame f(n+1) is increased to reduce the amount of computation made by the encoder 320 and the decoder 340. Accordingly, the decoder 340 may properly perform decoding. In this case, the amount of a DCT computation performed in the decoder 340 decreases, but the quality of the image deteriorates.

[0038] If the amount of computation that may be performed by the decoder 340 is 300 MIPS, and a computation amount actually consumed on the n-th frame f(n) is 250 MIPS, the number of skipped DCT operations is reduced to increase the amount of computation of the encoder 320 and the decoder 340 by the difference between the amount of computation that may be performed and the actual amount of computation, that is, a computation amount of about 50 MIPS, so that the decoder 340 may make full use of the amount of computation.

[0039] In contrast with the prior art in which a threshold value $T_{current}$ for DCT skipping on a current frame is fixed, in the present invention, a threshold value $T_{current}$ or T_{n+1} of the current frame f(n+1) is updated in consideration of the amount of decoding performed on the previous frame f(n) in a decoder, that is, a decoder complexity, which is received from the decoder via a backward channel. Thus, the amount of computation allocated to be performed by the decoder may be fully used, and an input bit stream may be properly encoded.

[0040] In a process to calculate the threshold value T_{n+1} of the current frame f(n+1), a DCT computation complexity may be expressed as shown in Equation 1, based on the characteristics that the DCT computation complexity is proportional to a reciprocal of the product of a threshold value and a quantization coefficient:

$$C = \frac{\upsilon}{TQ} \qquad \text{...(1)}$$

wherein C denotes the DCT computation complexity, T denotes the threshold value, and Q denotes the quantization coefficient.

[0041] In Equation 1, if the computation complexity, the threshold value, and the quantization coefficient of an n-th frame f(n) are C_n, T_n, and Q_n, respectively, υ is equal to $C_n\ T_n\ Q_n$, and a target threshold value T_{n+1} of an (n+1)th frame can be obtained from Equation 2:

$$T_{n+1} = \frac{\upsilon}{C_t Q_{n+1}} \qquad \text{...(2)}$$

wherein C_t denotes a target DCT computation complexity for the (n+1)th frame f(n+1).

[0042] Based on Equations 1 and 2, the target threshold value T_{n+1} of the (n+1)th frame may also be obtained from Equation 3:

$$T_{n+1} = T_n \frac{Q_n \, C_n}{C_t \, Q_{n+1}} \qquad \text{...(3)}$$

Equation 3 may be re-arranged to Equation 4:

$$T_{n+1} = T_n \frac{Q_n}{Q_{n+1}} (1 + \frac{C_n - C_t}{C_t}) \qquad \text{...(4)}$$

[0043] Equation 4 may also be rearranged to Equation 5 to provide a little flexibility:

$$T_{n+1} = T_n \frac{Q_n}{Q_{n+1}} (1 + \frac{C_n - C_t}{aC_t}) \qquad \text{...(5)}$$

wherein α denotes a parameter to control a convergence speed. Based on Equation 5, the target threshold value T_{n+1} of the current frame f(n+1) depends on the threshold value T_n at the previous frame f(n), the quantization coefficients Q_n and Q_{n+1} at the previous and current frames, a DCT computation complexity C_n of the previous frame f(n), and the target DCT computation amount C_t of the current frame f(n+1).

[0044] A method in which the DCT computation amount controller 336 determines the threshold value T_{n+1} of the current frame f(n+1) based on Equation 5 and the decoder complexity obtained by the decoder complexity calculator 342 will now be described.

[0045] First, the DCT computation amount controller 336 in the encoder 320 calculates a target DCT computation amount of the current frame f(n+1) based on decoder complexity information received from the decoder complexity calculator 342. In an embodiment of the present invention, the target DCT computation amount for the current frame f(n+1) is calculated according to Equation 6:

$$C_t = \left(\sum_{j=n}^{n-k+1} \frac{C_{allowed}(j)}{C_{total}(j)} \right) \times \sum_{j=n}^{n-k+1} C_{IDCT}(j) \qquad \text{...(6)}$$

[0046] The encoder 320 calculates the target DCT computation amount for the current frame f(n+1) using Equation 6. However, the encoder 320 may also calculate the target DCT computation amount of the current frame f(n+1) according to a predetermined equation, based on $C_{allowed}(n)$ and $C_{total}(n)$ received from the decoder complexity calculator 342.

[0047] The threshold value T_{n+1} of the current frame f(n+1) may be calculated by substituting the target DCT computation amount C_t for the current frame f(n+1) obtained by Equation 6 into Equation 5.

[0048] The DCT computation amount calculator 336 determines using Equation 7 whether a DCT operation is to be skipped for an i-th block of the current frame f(n+1), based on the calculated threshold value T_{n+1} of the current frame f(n+1), an SAD value for the i-th block, $SAD_{n+1,i}$, and a quantization parameter Q_{n+1} for the i-th block:

$$\frac{SAD_{n+1,i}}{Q_{n+1}} < T_{n+1} \qquad ...(7)$$

[0049] In other words, if the left side of Equation 7 is smaller than a right side as shown in Equation 7, the DCT unit 322 is set in a not-coded mode where a DCT operation is skipped, and accordingly, does not perform a DCT on the i-th block of the current frame f(n+1). If not, the DCT 322 performs a DCT on the i-th block of the current frame f(n+1).

[0050] FIG. 4 is a flowchart illustrating a method of controlling the amount of a DCT computation based on decoder complexity, according to an embodiment of the present invention. Referring to FIGS. 3 and 4, in operation 410, the DCT computation amount controller 336 of the encoder 320 receives decoder complexity information for k recently-decoded frames, $\{ C_{IDCT}(j), C_{allowed}(j)\ C_{total}(j) \mid j= n- k+1, . . . , n\}$, from the decoder complexity calculator 342 of the decoder 340 via the backward channel. The decoder complexity information of k recently-decoded frames includes information on a percentage of the total amount of computation performed in the decoder 340 occupied by the amount of IDCT computation, $C_{IDCT}(n)$, information on the amount of computation that may be performed by the decoder, $C_{allowed}(n)$, and information on the actual amount of computation $C_{total}(n)$. As described above, in the embodiment of the present invention, the IDCT computation amount $C_{IDCT}(j)$ is included in the decoder complexity information. However, the IDCT computation amount $C_{IDCT}(j)$ may be obtained by the IDCT unit 328 of the encoder 320.

[0051] In operation 420, the target DCT computation amount C_t of the current frame f(n+1) to be encoded is calculated based on the received decoder complexity information. In the embodiment of the present invention, Equation 6 is used to calculate the target DCT computation amount C_t of the current frame f(n+1). However, a different equation may be used.

[0052] In operation 430, the DCT computation amount controller 336 calculates the threshold value T_{n+1} used to skip a DCT operation on the current frame f(n+1), based on the target DCT computation amount C_t calculated in operation 420, the DCT computation amount information C_n of the previous frame f(n) received from the DCT unit 312, and the quantization coefficients Q_n and Q_{n+1} of the previous and current frames f(n) and f(n+1) received from the Q unit 314.

[0053] In operation 440, the DCT computation amount controller 336 determines with respect to each 8×8 block whether the DCT unit 312 skips performing a DCT, based on the SAD value of the i-th block of the current frame f(n+1), which is used to determine whether to skip execution of a DCT, operation and is received from the ME/MC unit 332, and the quantization coefficient Q_{n+1} received from the Q unit 314. To be more specific, if $SAD_{n+1,i}/ Q_{n+1}$ is smaller than the threshold value T_{n+1} of the current frame f(n+1), operation 440 proceeds to operation 450. On the other hand, if $SAD_{n+1,i}/ Q_{n+1}$ is equal to or greater than the threshold value T_{n+1} of the current frame f(n+1), operation 440 proceeds to operation 460.

[0054] In operation 450, the operation of the DCT unit 312 with respect to the i-th block of the current frame f(n+1) is set to be in a non-coded mode, and accordingly the DCT unit 312 does not perform a DCT operation on the i-th block of the current frame f(n+1). Operation 450 proceeds to operation 470.

[0055] In operation 460, the DCT unit 312 performs a DCT operation on the i-th block of the current frame f(n+1). Operation 460 proceeds to operation 470.

[0056] In operation 470, it is determined whether the i-th block is the last in the current frame. If it is determined in operation 470 that the i-th block is the last block in the current frame, operation 470 proceeds to operation 480. If the I-th block is not the last block in the current frame, operation 470 goes back to operation 440, and accordingly operations 440 through 470 are repeated.

[0057] In operation 480, it is determined whether input motion image data have been completely encoded. Accordingly, the above operations are repeatedly performed until all of the input motion image data are encoded.

[0058] Such an encoding in consideration of the amount of computation performed by a decoder enables a decoder to make full use of an allowed amount of computation.

[0059] FIG. 5 is a block diagram of a motion image encoder 520 according to another embodiment of the present invention and a decoder 540. Referring to FIG. 5, the encoder 520 includes the elements of the encoder 320 of FIG. 3 and further an ME computation amount calculator 538 to calculate a variation in the amount of an ME computation made by an ME/MC unit 532.

[0060] The ME computation amount calculator 538 calculates a difference between the target amount of an ME computation by the ME/MC unit 532, $C_{ME,target}$, and the amount of an ME computation actually performed on the previous frame f(n), $C_{ME}(n)$, and transmits ME computation amount variation information based on the calculated difference to a DCT computation amount controller 536.

[0061] Since elements other than the DCT computation amount controller 536 and the ME computation amount calculator 538 are the same as the elements of the same names of the encoder 320 of FIG. 3, the elements will not be described in detail.

[0062] FIG. 6 is a flowchart illustrating a method of controlling the amount of DCT computation in consideration of ME computation amount variation information and decoder complexity, according to another embodiment of the present invention.

[0063] Referring to FIGS. 5 and 6, in operation 610, the DCT computation amount controller 536 of the encoder 520 receives decoder complexity information from the decoder complexity calculator 542 via the backward channel and the ME computation amount variation information from the ME computation amount calculator 538. The decoder complexity information includes information on a ratio of the amount of processing of an IDCT computation to the total amount of computation performed in the decoder 540, $C_{IDCT}(n)$, information on the amount of computation that can be performed by the decoder, $C_{allowed}(n)$, and information on the actual amount of computation $C_{total}(n)$.

[0064] As described above, in the embodiment of the present invention, the IDCT computation amount $C_{IDCT}(j)$ is included in the decoder complexity information. However, the IDCT computation amount $C_{IDCT}(j)$ may be obtained by an IDCT unit 528 of the encoder 520.

[0065] In operation 620, the target DCT computation amount C_t of the current frame f(n+1) to be encoded is calculated based on the received decoder complexity information and ME computation amount variation information, using Equation 8:

$$C_t = \left(\sum_{j=n}^{n-k+1} \frac{C_{allowed}\ (j)}{C_{total}\ (j)} \right) \times \sum_{j=n}^{n-k+1} C_{IDCT}\ (j) + C_{ME,target} - C_{ME}\ (n) \qquad ...(8)$$

[0066] wherein $C_{ME,target}$ denotes the target amount of an ME computation performed on the current frame f(n+1) by the ME/MC unit 532, and $C_{ME}(n)$ denotes the amount of an ME computation actually performed on the previous frame f(n).

[0067] In the embodiment of the present invention, Equation 8 is used to calculate the target DCT computation amount C_t of the current frame f(n+1) to be encoded. However, a different equation may be used.

[0068] In operation 630, the DCT computation amount controller 536 calculates the threshold value T_{n+1} used to skip a DCT operation on the current frame f(n+1), based on the target DCT computation amount C_t calculated in operation 620, the DCT computation amount information C_n of the previous frame f(n) received from a DCT unit 522, and the quantization coefficients Q_n and Q_{n+1} of the previous and current frames f(n) and f(n+1) received from a Q unit 524.

[0069] In operation 640, the DCT computation amount controller 536 determines with respect to each 8×8 block whether the DCT unit 522 skips performing a DCT, based on the SAD value of the i-th block of the current frame f(n+1), which is used to determine whether to skip execution of a DCT operation and received from the ME/MC unit 532, and the quantization coefficient Q_{n+1} received from the Q unit 542. To be more specific, if $SAD_{n+1,i}/\ Q_{n+1}$ is smaller than the threshold value T_{n+1} of the current frame f(n+1), operation 640 proceeds to operation 650. On the other hand, if $SAD_{n+1,i}/\ Q_{n+1}$ is equal to or greater than the threshold value T_{n+1} of the current frame f(n+1), operation 640 proceeds to operation 660.

[0070] In operation 650, the operation of the DCT unit 522 with respect to the i-th block of the current frame f(n+1) is set to be in a non-coded mode, and accordingly the DCT unit 522 does not perform a DCT operation on the i-th block of the current frame f(n+1).

[0071] In operation 660, the DCT unit 522 performs a DCT operation on the i-th block of the current frame f(n+1).

[0072] In operation 670, it is determined whether the i-th block is the last block in the current frame. If it is determined in operation 670 that the i-th block is the last block in the current frame, operation 670 proceeds to operation 680. If the I-th block is not the last block in the current frame, operation 670 goes back to operation 640, and accordingly operations 640 through 670 are repeated.

[0073] In operation 680, it is determined whether input motion image data have been completely encoded. Accordingly, the above operations are repeatedly performed until all of the input motion image data are encoded.

[0074] As described above, by controlling the amount of DCT computation in consideration of the complexity of a computation made by the decoder 540 and the variation in the amount of a computation by the ME/MC unit 524, the decoder 540 may make full use of an allowed computation amount. Also, the total amount of computation performed in an encoder may be maintained close to a target amount of computation.

[0075] While the present invention has been particularly shown and described with reference to exemplary embodiments thereof, it will be understood by those of ordinary skill in the art that various changes in form and details may be made therein without departing from the spirit and scope of the present invention as defined by the following claims.

[0076] The invention may also be embodied as computer readable codes on a computer readable recording medium. The computer readable recording medium is a data storage device that may store data which can be thereafter read by a computer system. Examples of the computer readable recording medium include read-only memory (ROM), random-access memory (RAM), CD-ROMs, magnetic tapes, floppy disks, optical data storage devices, and the like. Also, the computer readable codes may be transmitted via a carrier wave such as the Internet. The computer readable recording medium may also be distributed over a network coupled computer systems so that the computer readable code is stored and executed in a distributed fashion.

[0077] As described above, in a DCT computation amount controlling method and apparatus according to the present invention, a motion image encoder employs a DCT skipping technique that considers information on the amount of a computation by a decoder and a variation in the amount of a computation by an ME unit. Thus, a computation amount allowed to the decoder may be maximally used, and an input bit stream may be normally encoded. Also, the computation complexity of the motion image encoder may be constantly maintained regardless of the characteristics of an image.

[0078] Although a few embodiments of the present invention have been shown and described, it would be appreciated by those skilled in the art that changes may be made in this embodiment without departing from the principles and spirit of the invention, the scope of which is defined in the claims and their equivalents.